Middle East Technical University

Department of Computer Engineering

CENG 111

Fall 2018

Take Home Exam 2

REGULATIONS

Due date: 10 December 2016, Wednesday, 23:59 *(Not subject to postpone)*

Submission: Electronically. You should save your program source code as a text file named `the2.py` and submit it to us via the course's COW page.

Team: There is **no** teaming up. This is an EXAM.

Cheating: Source(s) and Receiver(s) will receive zero and be subject to disciplinary action.

INTRODUCTION

A *Firmus* is a stable stacking of two rectangular blocks. Below you see some example firmuses.

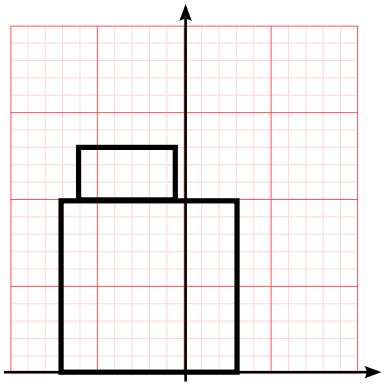


Figure 1

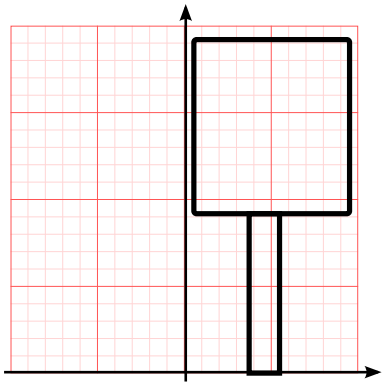


Figure 2

Although a firmus can be thought to exist in higher dimensions with more sophisticated blocks, we will assume that the blocks have homogeneous mass distribution and are two dimensional. Therefore, boxes can be assumed to be on a plane that is spanned by a Cartesian coordinate system spanned by x and y axes. We will also assume that gravity acts in the $-y$ direction. Blocks' edges are always parallel to the axes of the coordinate system.

For two blocks to form a firmus, the following conditions should be met:

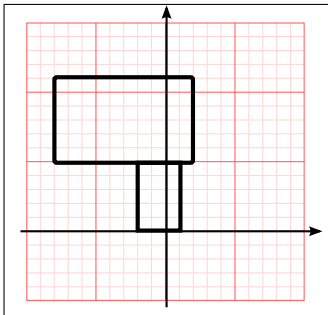
I. The lower block should have its lower edge placed directly on the x axis.

II. The upper block should have its lower edge (at least partially) coincide with the upper edge of the lower block.

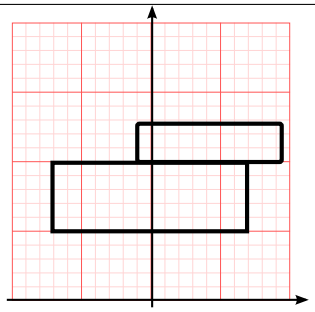
III. The upper block should have its center of mass abscissa in the range of the lower block's upper edge.

Anything else is **not** a firmus.

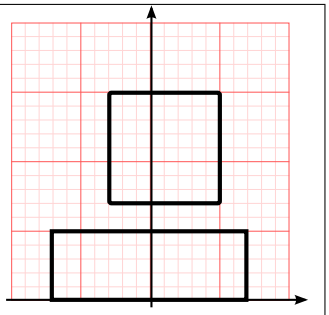
Below are some examples which do not form a firmus:



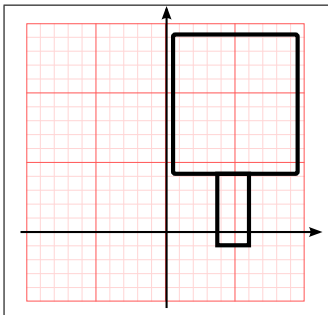
Violates **III**
Figure 3



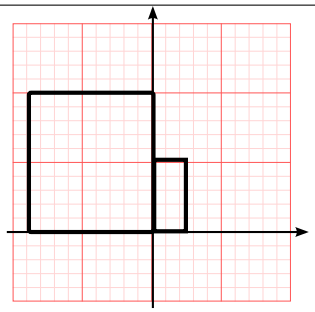
Violates **I**
Figure 4



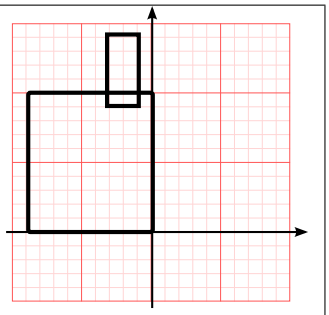
Violates **II**
Figure 5



Violates **I**
Figure 6



Violates **II**
Figure 7



Violates **II**
Figure 8

PROBLEM & SPECIFICATIONS

- You are expected to write a function `is_firmus` which takes two lists describing the two blocks.
- A block will be provided as the following list: $[x_1, y_1, x_2, y_2]$, where (x_1, y_1) and (x_2, y_2) are the diagonal corners of the block. However, it is not known whether you are given the (lower-left & upper-right) corners, or the (upper-left & lower-right) corners.
- The `is_firmus` function should return one of the following lists:
  - `["FIRMUS", area]` if the two blocks form a firmus. `area` is a floating point number for the area of the firmus.
  - `["ADDENDUM", [x1, y1, x2, y2]]` if the two blocks do not form a firmus but the first two criteria to be a firmus hold. `[x1, y1, x2, y2]` are the diagonal coordinates of the smallest third block which, when created and glued to the upper block, makes the lower block and the extended upper block a firmus. With the extension, the upper block should still be a rectangle.
  - `["DAMNARE", area]` in all other cases. `area` is a floating point number for the total area covered by both blocks. Note that this area is not necessarily equal to the sum of the two block's areas since an overlapping area, if any, should be counted only once.

EXAMPLES

Figure	Possible function call	Return value
Figure 1	`is_firmus([-0.5,10,-6,13],[-7,0,3,10])`	`['FIRMUS', 116.5]`
Figure 2	`is_firmus([0.5,19,9.5,9],[3.8,9,5.5,0])`	`['FIRMUS', 105.3]`
Figure 3	`is_firmus([-8,11,2,5],[1,0,-2,5])`	`['ADDENDUM', [2, 5, 4, 11]]`
Figure 4	`is_firmus([-7,5,7,10],[9.5,12.6,-1,10])`	`['DAMNARE', 97.3]`
Figure 5	`is_firmus([-3,7,5,15],[-7,0,7,5])`	`['DAMNARE', 134.0]`
Figure 6	`is_firmus([6,4,3.9,-1],[0.5,14.2,9.5,4])`	`['DAMNARE', 102.3]`
Figure 7	`is_firmus([0,0,2.4,5.2],[-8.7,10,0,0])`	`['DAMNARE', 99.48]`
Figure 8	`is_firmus([0,10,-8.7,0],[-4,9,-1,14])`	`['DAMNARE', 99.0]`

As far as the six non-firmus examples above (Figures 3-8) are concerned: All except the first one are not fixable. Therefore, they will be categorized as `DAMNARE`. The first one, though, is a violation of **III** only, which is fixable by the following `ADDENDUM`:

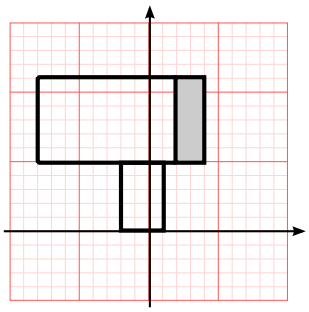


Figure 9

Note that, for this case, there are other representation alternatives for the added block. You are free to pick any.

NOTES

- Two floats f_1 and f_2 will be assumed to be the same provided that $|f_1 - f_2| < 0.001$ is true.
- All corner coordinates will remain in the range $[-100.0, +100.0]$.
- Degenerate (i.e., erroneous) test cases will not be considered.
- There are no unstable balance cases (i.e., cases where the upper block is on the verge of falling down and barely standing on the lower block). Any such case shall be considered as a 'stable balance', as far as the allowed difference between floats is concerned.
- Regularly follow any further announcements or specifications, and post your questions on the newsgroup of the course, if any.

GRADING

- Comply with the specifications. Since your returned results will be evaluated automatically, non-compliant results will be considered as incorrect by our evaluation system.
- Your program will be tested with multiple data (a distinct run for each data).

- Any program that performs only 30% and below will enter a glass-box test (eye inspection by the grader TA). The TA will judge an overall THE2 grade in the range of [0,30]. The glass-box test grade is not open to negotiation nor explanation.
- A program based on randomness will be graded zero.